

The Effect of Exerting Adequate Persistence in Collision Avoidance Protocols

J.J. GARCIA-LUNA-ACEVES

jj@cse.ucsc.edu

Computer Engineering Department
University of California
Santa Cruz, California 95064
Networking and Security Center
Sun Microsystems Laboratories
Palo Alto, California 94303

ASIMAKIS TZAMALOUKAS

jamal@cse.ucsc.edu

Computer Engineering Department
University of California
Santa Cruz, CA 95064

Abstract—Many medium-access control (MAC) protocols based on a collision-avoidance handshake between the sender and the receiver have been proposed for wireless networks. To date, however, the analysis of these protocols has assumed non-persistent strategies in sending control packets for collision avoidance. The persistent strategies discussed in the past for CSMA and CSMA/CD provide performance improvements over non-persistent access only at small traffic loads. We present and analyze a limited persistence approach to the transmission of collision-avoidance control packets. With limited persistence, a node senses the channel before sending collision-avoidance control packets. If the channel is sensed busy, the node persists sensing for an amount of time proportional to the transmission time of a control packet. The node can transmit its control packet if the channel is idle within its persistence waiting time and the channel is known to be available for transmissions; otherwise, the node backs off for a random amount of time and tries sending its control packet at the end of that time. We analyze the effect of limited persistence in source-initiated and receiver-initiated collision avoidance protocols by comparing their throughput with and without persistence; the analysis shows that limited persistence makes collision-avoidance protocols more efficient.

I. INTRODUCTION

There is a large body of work on the design of MAC (medium access control) protocols for wireless networks with hidden terminals. Kleinrock and Tobagi [7] identified the hidden-terminal problem of carrier sensing, which makes carrier-sense multiple access (CSMA) perform as poorly as the pure ALOHA protocol when the senders of packets cannot hear one another and the vulnerability period of packets becomes twice a packet length. The BTMA (busy tone multiple access) protocol was a first attempt to solve the hidden-terminal problem by introducing a separate busy tone channel [11]. The same authors proposed SRMA (split-channel reservation multiple access) [12], which attempts to avoid collisions by introducing a control-signal handshake between the sender and the receiver. A station that needs to transmit data to a receiver first sends a request-to-send (RTS) packet to the receiver, who responds with a clear-to-send (CTS) if it receives the RTS correctly. A sender transmits a data packet only after receiv-

This work was supported in part at UCSC by the Defense Advanced Research Projects Agency (DARPA) under Grant No. F30602-97-2-0338.

ing a CTS successfully. ALOHA or CSMA can be used by the senders to transmit RTSs.

Several variations of this scheme have been developed since SRMA was first proposed, including MACA [6], IEEE 802.11 [1], and FAMA-NCS (floor acquisition multiple access with non-persistent carrier sensing) [3]. These examples, and most protocols based on collision-avoidance handshakes to date are sender-initiated, in that the node wanting to send a data packet first transmits a short RTS asking permission from the receiver. More recently, however, receiver-initiated collision avoidance protocols have been proposed in which the receivers poll the senders for data packets; examples of these type of collision avoidance protocols are MACA-BI (multiple access collision avoidance by invitation) [9] and RIMA (receiver initiated multiple access) [5]. A receiver-initiated collision avoidance strategy is attractive because it can reduce the number of control packets needed to avoid collisions.

All comparative performance analysis to date for both sender- and receiver-initiated collision-avoidance protocols [2], [3], [5], [4], [9] have assumed non-persistent channel access for the transmission of collision-avoidance control packets. With a non-persistence approach to collision-avoidance, a node senses the channel before transmitting collision-avoidance control packets. If the channel is sensed idle, the node transmits its control packet; otherwise, the node backs off for a random amount of time and attempts to transmit at that later time.

The use of persistence in MAC protocols has been reported for CSMA [7], [8] and CSMA/CD [10]. The persistence strategies reported in the literature consist of a node with a packet to send that senses the channel being busy to persist with certain probability in sending its packet as soon as the channel is sensed idle again. As traffic load increases in the channel, the likelihood that many nodes will try to transmit immediately at the end of an ongoing transmission increases substantially, which makes traditional persistent CSMA and CSMA/CD unattractive for networks without light traffic loads.

We introduce a new persistence strategy aimed at collision

Report Documentation Page			<i>Form Approved OMB No. 0704-0188</i>	
<p>Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.</p>				
1. REPORT DATE 1999	2. REPORT TYPE	3. DATES COVERED 00-00-1999 to 00-00-1999		
4. TITLE AND SUBTITLE The Effect of Exerting Adequate Persistence in Collision Avoidance Protocols			5a. CONTRACT NUMBER	
			5b. GRANT NUMBER	
			5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)			5d. PROJECT NUMBER	
			5e. TASK NUMBER	
			5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) University of California at Santa Cruz, Department of Computer Engineering, Santa Cruz, CA, 95064			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)			10. SPONSOR/MONITOR'S ACRONYM(S)	
			11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited				
13. SUPPLEMENTARY NOTES The original document contains color images.				
14. ABSTRACT				
15. SUBJECT TERMS				
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES 10
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified		
19a. NAME OF RESPONSIBLE PERSON				

avoidance MAC protocols that limits the contention among nodes that receive packets to send at the time the channel is busy. The limited persistence mechanism we introduce is very simple and consists of establishing a time bound on how long a node can persist transmitting once it has a packet to send and sensed the channel busy. More specifically, when a node receives a packet to send (control packet or data packet depending on the protocol), it senses the channel. If the channel is sensed to be idle and no other node has the right to transmit, the node transmits its packet; otherwise, the node persists trying to send its packet for a *persistence time* of η seconds, which by design is much smaller than a data packet and in the case of collision-avoidance protocols is proportional to the transmission time of a control packet. If the channel becomes idle before the persistence time of the node elapses, the node transmits its packet if no other node has the right to transmit; otherwise, the node backs off for a random amount of time and attempts to transmit at that later time.

The objective of introducing a limited window of persistence in collision-avoidance protocols is twofold. Allowing some degree of persistence, protocol performance is improved at light loads because it reduces the number of times in which a single node with a packet to send after sensing the channel busy must back off for a relatively long time. At the same time, limiting the amount of time any node can persist transmitting after detecting a busy channel improves performance relative to traditional persistent strategies, because it reduces the amount of contention at the time the channel becomes idle. The main contribution of this paper consists of showing that persistence in collision avoidance can be beneficial to the performance of the system, provided that the adequate amount of persistence is applied.

Section II describes sender- and receiver-initiated protocols with limited persistence; we modify FAMA-NCS [3] and RIMA protocols [5] to operate with limited persistence, because they have been shown to be the best performing sender-initiated and receiver-initiated collision-avoidance protocols with non-persistent carrier sensing. Section III uses an analytical model to study the throughput of these protocols in fully-connected networks and compares the performance of the protocols with non-persistence and limited persistent carrier sensing. We use a fully-connected network topology to discern the relative performance advantages of different protocols, because of two reasons: (a) it allows us to use a short analysis that can be applied to several protocols, and (b) our focus on protocols that provide correct collision avoidance means that the relative performance differences in a fully-connected network are very much the same when networks with hidden terminals are considered.

II. COLLISION AVOIDANCE PROTOCOLS

Carrier sensing has been shown to increase the throughput of sender-initiated and receiver-initiated collision avoidance and to be necessary to avoid collisions of data packets with other packets at the receivers [2], [5] in single-channel networks.

The rest of this section describes sender- and receiver-initiated collision-avoidance protocols with limited-persistence carrier sensing (LCS). The proofs that these protocols support correct collision avoidance in the presence of hidden terminals are essentially the same as those published for the non-persistent versions of the protocols.

A. Sender-Initiated Protocols

In sender-initiated collision avoidance we describe a variant of FAMA-NCS, which is based on non-persistent carrier sensing. This variant is called FAMA-LCS (limited-persistence carrier sensing), and its operation on a fully-connected network is depicted in Fig. 1.

In FAMA-LCS, the sender of a packet transmits a short Request-To-Send (RTS) packet asking the receiver permission to transmit. To send its RTS, the sender uses LCS. More specifically, if the sender senses the channel to be idle and no other node has the floor (right to transmit), the sender transmits its RTS. Alternatively, if the sender senses a busy channel, it persists trying to transmit its RTS for a persistence time of η seconds equal to or smaller than the transmission time of an RTS (γ). If the channel becomes idle before the persistence time elapses, the sender transmits its packet, unless another node has the right to transmit on the channel; otherwise, the sender backs off for a random amount of time and attempts to transmit its RTS at a later time.

Once an RTS is sent, the receiver responds to a correctly addressed RTS with a Clear-To-Send (CTS) packet. The sender transmits its data packet upon reception of the CTS, and the receiver sends an acknowledgment to a correctly received data packet.

As in FAMA-NCS, the length of a CTS in FAMA-LCS equals the length of an RTS plus at least a maximum round-trip delay in order to ensure correct collision avoidance [3].

As Fig. 1 illustrates, a successful RTS can occur when a node receives a packet to send when the channel is idle, as well as when the channel is busy, provided that the channel becomes available within η seconds from the arrival of the packet to be sent. Similarly, an RTS can fail when multiple RTSs are sent within τ seconds of one another when the channel is idle, or when multiple RTSs are scheduled for transmission within the last η seconds of a transmission period, after which the channel becomes available.

B. Receiver-Initiated Protocols

In receiver-initiated collision-avoidance protocols, the receivers poll the senders for packets to be sent. We assume that this polling is data driven, in which a node attempts to poll its neighbors at a rate that is a function of the data rate with which it receives data to be sent, as well as the rate with which the node hears its neighbors send control and data packets. We present variants of RIMA protocols that incorporate LCS and differ on the type of polling packets sent by the receivers.

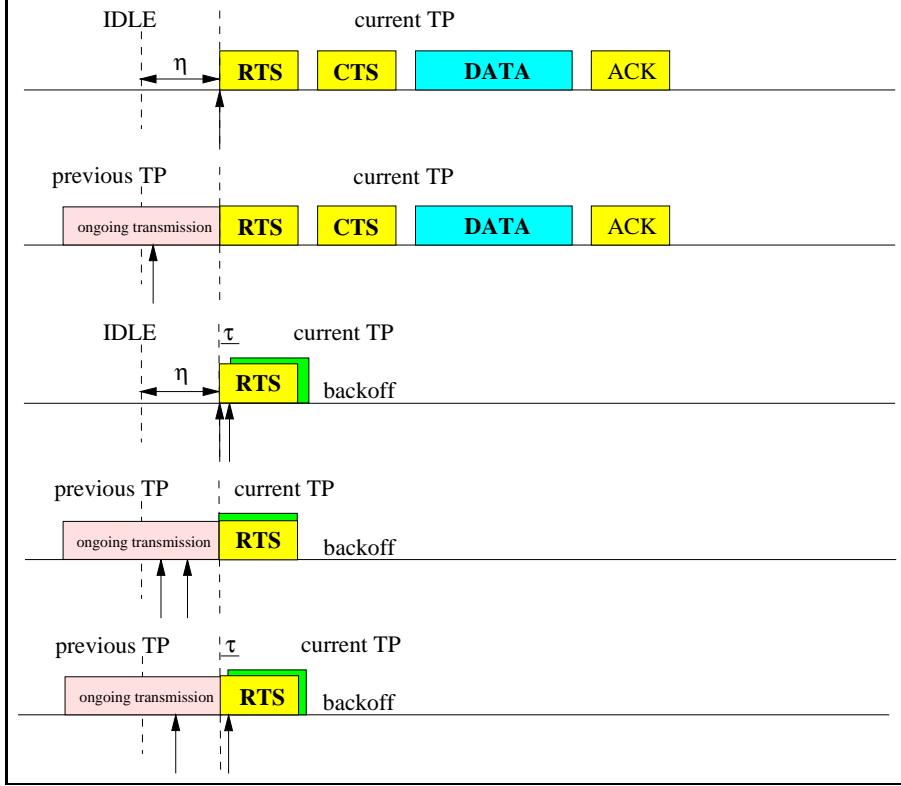


Fig. 1. FAMA-LCS illustrated

B.1 RIMA with Simple Polling

In the RIMA-SP (simple polling) protocol [5], the receiver sends a ready-to-receive (RTR) packet to a particular sender. If the polled node has data to send, it waits for a *collision avoidance period* of length ξ that allows the polling node to abort the transaction after detecting noise in the channel by sending a no-transmission-request (NTR) packet. If the polled node perceives the channel idle during the collision avoidance period, it transmits its data packet to the polling node if it has any packets intended for it.

We modify RIMA-SP by making the polling node use LCS for the transmission of its RTRs. We call the resulting variant RIMA-SPL (simple polling with limited persistence). In RIMA-SPL, the polling node senses the channel before sending its RTR, and transmits the RTR if the channel is idle and no other node has gained control of the channel. If the polling node senses a busy channel, it persists for a persistence time lasting η seconds, which is equal to or smaller than the length of an RTR (γ). If the polling node senses that the channel becomes idle and no other node attains control of the channel before the persistence time elapses, it transmits its RTR; otherwise, it backs off for a random amount of time and tries to send its RTR at a later time. Fig. 2 illustrates the operation of RIMA-SPL for a fully-connected network. Like FAMA-LCS, a successful RTR occurs when the channel is idle or becomes idle in less than η seconds from the time that a local packet has arrived. However, since RTRs are not always followed by data

in RIMA-SPL we can have the two additional failed periods of Fig. 2(c).

In RIMA-SPL, every node initializes itself in the START state, in which the node waits twice the maximum channel propagation delay, plus the hardware transmit-to-receive transition time (ϵ), before sending anything over the channel. This enables the node to find out if there are any ongoing transmissions. After a node is properly initialized, it transitions to the PASSIVE state. In all the states, before transmitting anything to the channel, a node must listen to the channel for a period of time that is sufficient for the node to start receiving packets in transit.

If a node x is in the PASSIVE state and senses carrier, it transitions to the REMOTE state to defer to ongoing transmissions. A node in REMOTE state must allow enough time for a complete successful handshake to take place, before attempting to transition from remote state.

Any node in PASSIVE state that detects noise in the channel must transition to the BACKOFF state. If node x is in PASSIVE state and obtains an outgoing packet to send to neighbor z , it transitions to the RTR state. In the RTR state, node x uses LCS to transmit an RTR. If node x detects carrier when it attempts to send the RTR, it starts a persistence timer lasting η seconds. If the channel remains busy during the η seconds or the channel becomes idle but another node gains the right to use the channel, the node transitions to the BACKOFF state. This step makes the node back off immediately for a suf-

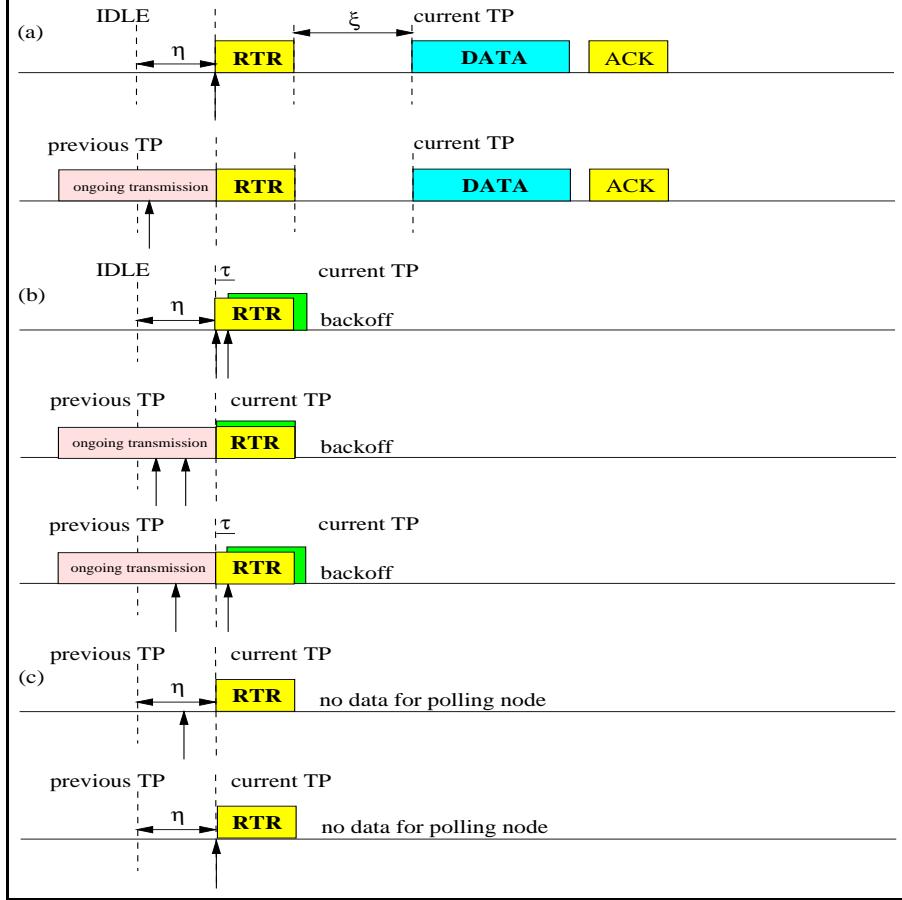


Fig. 2. RIMA-SP with limited persistent carrier sensing

ficient amount of time to allow a complete handshake between a sender-receiver pair to occur; otherwise, x sends its RTR. If the node detects an idle channel and no node gains control of the channel before the η seconds of the persistence timer expire, the node transmits its RTR.

If node z receives the RTR correctly and has data for x , it waits for ξ seconds. If during the waiting period there is no activity in the channel, node z transitions to the XMIT state, where it transmits a data packet to x and node x sends an acknowledgment (ACK) immediately after receiving the data packet (Fig. 2(a)); otherwise, node z assumes that there was a collision and transitions to the BACKOFF state to allow floor acquisition by some other node. After sending its RTR, node x senses the channel. If it detects carrier immediately after sending its RTR, node x assumes that a collision or a successful data transfer to a hidden node is taking place. Accordingly, it sends a No Transmission Request (NTR) to z to stop z from sending data that would only collide at x . This scenario can only occur in a multi-hop network topology.

When multiple RTRs are transmitted within a one-way propagation delay a collision takes place and the nodes involved have to transition to the BACKOFF state and try again at a later time chosen at random, as shown in Fig. 2(b).

Node x determines that its RTR was not received correctly by z after a time period equal to the maximum round-trip delay to its neighbors plus turn-around times and processing delays at the nodes, plus the waiting period ξ . After sending its RTR, node x listens to the channel for any ongoing transmission. Because of non zero propagation delays, if node x detects carrier immediately after transmitting its RTR, it can conclude that it corresponds to a node other than z , which would take a longer time to respond due to its need to delay its data to x to account for turn-around times.¹

The lengths of RTRs and NTRs are the same. The same argument used in [2] to show that the length of an RTS must be longer than the maximum propagation delay between two neighbors to ensure correct collision avoidance can be used to show that RTRs and NTRs must last longer than a maximum propagation delay. In ad-hoc networks in ISM bands, propagation delays are much smaller compared with any packet that needs to be transmitted.

To reduce the probability that the same nodes compete repeatedly for the same receiver at the time of the next RTR, the RTR specifies a back off period unit for contention. The

¹Our analysis assumes 0 turn-around times and 0 processing delays for simplicity.

nodes that must enter the BACKOFF state compute a random time that is a multiple of the back off-period unit advertised in the RTR. The simplest case consists of computing a random number of back off-period units using a uniformly distributed random variable from 1 to d , where d is the maximum number of neighbors for a receiver. The simplest back off-period unit is the time it takes to send a small data packet successfully.

B.2 RIMA with Dual-Use Polling

RIMA-DP (dual-purpose polling) [5] improves over RIMA-SP by making the RTR into a request for data from the polled node, as well as a transmission request for the polling node to send data. We refer to the variant of RIMA-DP with limited-persistence carrier sensing by RIMA-DPL (dual-purpose polling with limited persistence). Fig. 3 illustrates the operation of this protocol. With RIMA-DPL, a successful RTR can be followed from one or two data packet transmission as shown in Fig. 3(a).

A key benefit of the dual-use polling in RIMA-DP and RIMA-DPL is that both polling and polled nodes can send data in a round of collision avoidance. This is possible because the RTR makes all the neighbors of the polling node back off, and the data from the polled node make all its neighbors back off, which can then be used by the polling node to send its data.

In RIMA-DPL, a sender with an RTR to be sent senses the channel, and transmits the RTR if the channel is idle and no other node has control of the channel. If the channel is busy, the sender (polling node) persists trying to send the RTR for a persistence time of length η seconds that is smaller than or equal to the length of an RTR.

RIMA-DPL gives transmission priority to the polling nodes. When a node z is polled by node x and has data for node x , z waits for a collision avoidance period of ξ seconds before sending a data packet. In contrast, if the polled node does not have data for x , it immediately sends a CTS (Clear-To-Send packet) to x . This permits a polling node x exposed to a neighbor sending data to hear part of that neighbor's data packet after sending its RTR; in such a case, node x can send an NTR to the polled node to cancel its RTR. In [5] it is proven that RIMA-DP and consequently RIMA-DPL prevents collisions of data packets with any types of packets, provided that z waits for $\xi > \gamma + 7\tau$ seconds before sending any data after being polled and the length of a CTS is 2τ seconds longer than the length of an RTS. The lengths of RTRs and RTSs are the same.

Every node starts in the START state and transitions to the PASSIVE state when it is initialized. If a node x is in the PASSIVE state and senses carrier, it transitions to the REMOTE state to defer to ongoing transmissions. A node in REMOTE state must allow enough time for a complete successful handshake to take place, before attempting to transition from remote state.

Any node in PASSIVE state that detects noise in the channel must transition to the BACKOFF state where it must allow sufficient time for complete successful handshakes to occur. If node x is in PASSIVE state and obtains an outgoing packet to

send to neighbor z , it transitions to the RTR state. In the RTR state, node x behaves as in RIMA-SPL.

If node z receives the RTR correctly and has data for x , it waits for ξ seconds before sending a data packet to x . If during the waiting period there is no activity in the channel, node z transitions to the XMIT state, where it transmits a data packet to x . Otherwise, z assumes a collision or data transfer to a hidden node and goes to the BACKOFF state. If z has no data for x , it sends a CTS to x immediately.

If node x detects carrier immediately after sending an RTR, it defers its transmission attempt and sends an NTR to the node it polled. The CTS length, which is τ seconds longer than an RTR, forces polling nodes that send RTRs at about the same time when a polled node sends a CTS to detect carrier from the CTS and stop their attempt to send or receive data. Any node other than x receiving the CTS for x transitions to the BACKOFF state. When node x receives the CTS from z , it transitions to the XMIT state and transmits a data packet to z .

III. THROUGHPUT ANALYSIS

The objective of our analysis is to compare the performance of receiver- and sender-initiated protocols with limited persistence carrier sensing and non-persistent carrier sensing. The objective of the model we use is to analyze the effect of persistence on the throughput of the system. Because the protocols we analyze ensure correct collision avoidance in the presence of hidden terminals [3], [5], the relative differences in performance among these protocols are the same with and without hidden terminals [3]; accordingly, to simplify our model, we assume a fully-connected network.

A. Modeling Assumptions

Our modeling of limited-persistent carrier-sensing MAC protocols is based on the model first introduced by Sohraby et al. [8] which requires the following assumptions to be made:

1. There are N nodes in the fully-connected network.
2. A single unslotted channel is used for all packets, and the channel introduces no errors.
3. All nodes can detect collisions perfectly.
4. The size for a data packet is δ seconds and the size of an RTR, an ACK, and an RTS is γ seconds, the size of a CTS in RIMA protocols is γ seconds, and the size of a CTS for FAMA-NCS is $\gamma + 2\tau$ [3].
5. The turn-around time is considered to be part of the duration of control and data packets.
6. The propagation delay of the channel between any two nodes is τ seconds.
7. The collision avoidance interval used in RIMA protocols is ξ seconds.
8. The persistence timer in RIMA protocols is η seconds.

To provide a fair comparison between sender-initiated and receiver-initiated protocols while preserving the tractability of the analytical model, we assume that a polled node receiving an RTR always has a data packet to send, but the probability that that packet is addressed to the polling node is $\frac{1}{N}$. Furthermore,

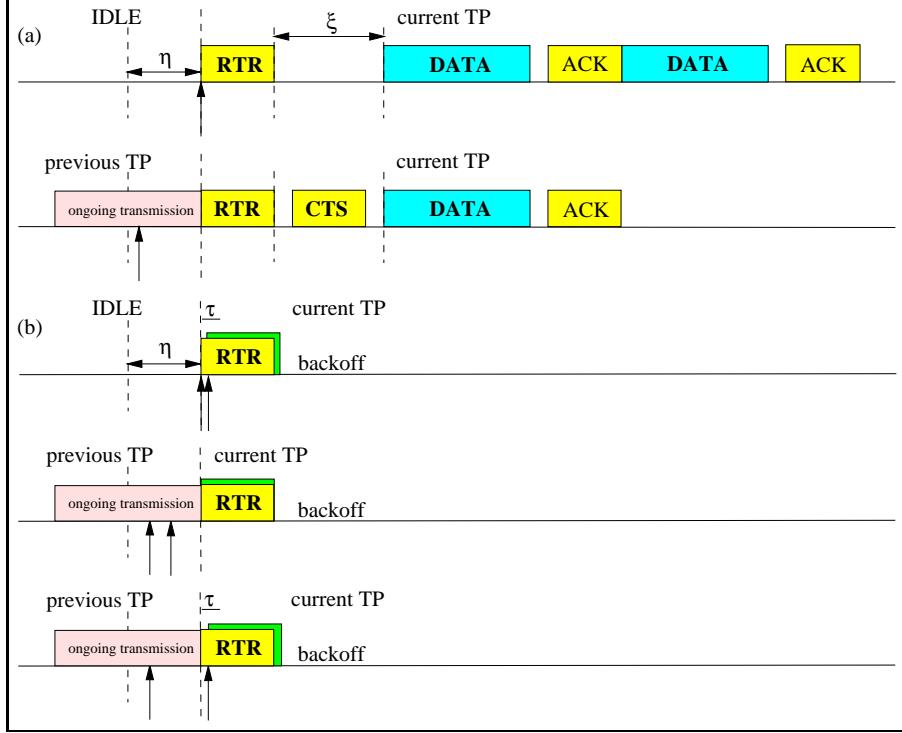


Fig. 3. RIMA-DP with limited persistent carrier sensing

we assume that each node sends its RTR according to a Poisson distribution with a mean rate of $\frac{\lambda}{N}$, and that (when applicable) the polling node chooses the recipient of the RTR with equal probability.

The corresponding assumptions for sender-initiated protocols are that a node always has packets to send, but schedules the transmission of RTSs according to a Poisson distribution with a mean rate of $\frac{\lambda}{N}$, and chooses to which neighbor to send the RTS with probability $\frac{1}{N}$. These assumptions preserve the validity of prior analytical results for FAMA and sender-initiated collision-avoidance MAC protocols [3].

B. Analysis

As Fig. 4 illustrates, under steady-state operation, the utilization of the channel consists of cycles of idle periods followed by busy periods.

A busy period consists of a sequence of one or more transmission periods, and each transmission period starts with the transmission of either one or multiple control packets (RTRs for RIMA and RTSs for FAMA). We define a transmission period of type 1 (TP 1 in Fig. 4) to be a transmission period that starts with a single RTR or RTS. We also define a transmission period of type 2 (TP 2 in Fig. 4) to be a transmission period that starts with the simultaneous transmission of two or more RTRs or RTSs. For convenience, we refer to idle periods as transmission periods of type 0 (TP 0 in Fig. 4).

Because the arrivals of RTRs or RTSs follow a Poisson distribution, a transmission period following a transmission period of type 0 is always of type 1. Because a node persists trying to

transmit an RTS or RTR for η seconds after detecting a busy channel, the type of transmission period that follows a transmission period of type 1 or 2 is defined solely by the number of RTS or RTR arrivals that occur during the last η seconds of the current transmission period.

Following the analysis by Sohraby et al. [8], we define the state of the system at the beginning of a transmission period to be the type of that transmission period. Because the type of transmission period reached depends only on the number of arrivals in the prior transmission period, the three possible states of the system and the possible transitions between them, correspond to a three-state Markov chain embedded at the beginning of the transmission periods. Fig. 5 illustrates this Markov chain.

No packets are transmitted during a type-0 transmission period. In contrast, during a type-2 transmission period, multiple RTRs or RTSs collide. A type-1 transmission period can be successful or unsuccessful, depending on the number of arrivals that occur during the vulnerability period of the transmission period. Furthermore, for the case of RIMA protocols, the length of a type-1 transmission period further depends on the availability of a packet for the polling node at the polled node. The vulnerability period of a transmission period is equal to the propagation delay of τ seconds needed for all nodes to detect the transmission of RTRs or RTSs that start the transmission period.

The transition probability from state i to state j is denoted by P_{ij} . We denote by π_i ($i = 0, 1, 2$) the stationary probability of being in state i , i.e., that the system is in a type- i transmission

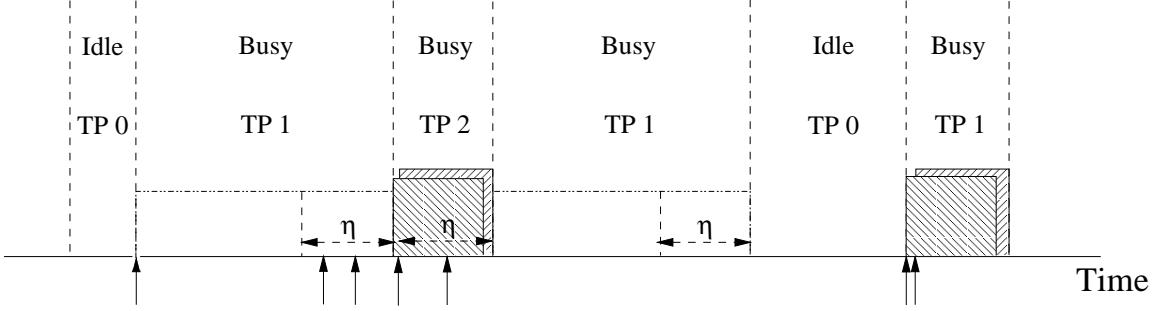


Fig. 4. 1-persistent transmission periods

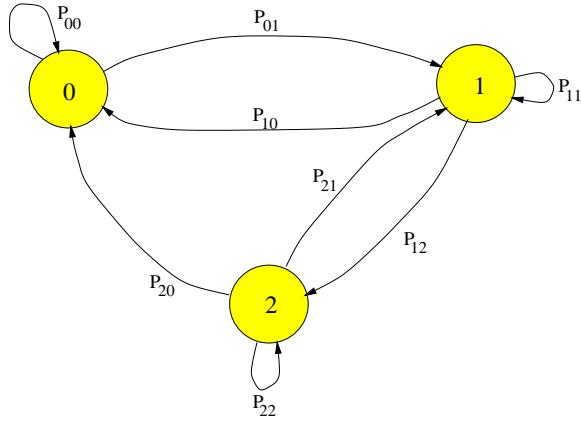


Fig. 5. State transitions of collision-avoidance protocols with limited persistence

period, and by T_i the average duration of the random variable that represents the length of a type- i transmission period. From renewal theory, we can define the throughput of the network by:

$$S = \frac{\pi_1 U}{\sum_{i=0}^2 \pi_i T_i} \quad (1)$$

where U is the average time during which data packets are sent in a successful transmission period.

To compute S we need to compute the state probabilities and the average duration of transmission periods. From the Markov state diagram we have the following four equations:

$$\begin{aligned} \pi_0 &= \pi_1 P_{10} + \pi_2 P_{20} \\ \pi_1 P_{12} &= \pi_2 (P_{21} + P_{20}) \\ \pi_0 + \pi_1 + \pi_2 &= 1 \\ P_{10} + P_{11} + P_{12} &= 1 \end{aligned} \quad (2)$$

The probability of transitioning to a type-0 transmission period from a type-1 or type-2 transmission period equals the probability that no RTR or RTS arrives during the last η seconds of the transmission period. The probability of transitioning to a type-1 transmission period from a type-1 or type-2 transmission period equals the probability that a single arrival of an RTR or an RTS takes place during the last η seconds

of the transmission period. Similarly, the probability of transitioning from a type-1 or type-2 transmission period to a type-2 transmission period equals the probability that two or more RTRs or RTSs arrive during the last η seconds of the prior transmission period. Accordingly, we have

$$P_{1j} = P_{2j} \quad j = 0, 1, 2 \quad (3)$$

The state probabilities can then be obtained from Eqs. (2) and (3) to be [8]:

$$\pi_0 = \frac{P_{10}}{1 + P_{10}} \quad (4)$$

$$\pi_1 = \frac{P_{10} + P_{11}}{1 + P_{10}} \quad (5)$$

$$\pi_2 = \frac{1 - P_{10} - P_{11}}{1 + P_{10}} \quad (6)$$

To compute the transition probabilities P_{11} and P_{10} , let Y denote the random variable representing the arrival time of the last RTR or RTS that arrives during the vulnerability period of a transmission period. Conditioning on $Y = y$, a transition from a type-1 transmission period to a type-0 transmission period occurs if there are no arrivals of RTRs or RTSs in the time interval spanning the last η seconds of the type-1 transmission period and the first y seconds of the type-0 transmission period. Given that the arrival of RTRs or RTSs is Poisson with parameter λ , the probability of this event equals $e^{-\lambda(\eta+y)}$. Unconditioning, we have [8]:

$$P_{10} = (1 + \lambda\tau)e^{-\lambda(\eta+\tau)} \quad (7)$$

Following the same approach, we find that

$$P_{11} = \lambda e^{-\lambda(\eta+\tau)} [\eta + \lambda\tau(\eta + \tau/2)] \quad (8)$$

Substituting Eqs. (7), (8), and (4) to (6) in Eq. (1), we obtain that the throughput of the system equals

$$\begin{aligned} S &= \frac{U(P_{10} + P_{11})}{T_0 P_{10} + T_1 (P_{10} + P_{11}) + T_2 (1 - P_{10} - P_{11})} \\ &= \frac{U(1 + \lambda\tau + \lambda[\eta + \lambda\tau(\eta + \tau/2)])}{\{T_0(1 + \lambda\tau) + T_1(1 + \lambda\tau + \lambda[\eta + \lambda\tau(\eta + \tau/2)]) + T_2(e^{\lambda(\eta+\tau)} - 1 - \lambda\tau - \lambda[\eta + \lambda\tau(\eta + \tau/2)])\}} \end{aligned} \quad (9)$$

The throughput achieved by each collision-avoidance protocol can now be obtained as a function of the rate of arrival of RTRs or RTSs in the system by obtaining the values of U , T_0 , T_1 , and T_2 for each protocol.

The throughput of collision-avoidance protocols is specified in Theorems 1 to 3 below, making use of the following definitions:

$$A = e^{\lambda\tau}(\gamma + 2\tau - 1/\lambda)$$

$$B = 1 + \lambda\tau + \lambda[\eta + \lambda\tau(\eta + \tau/2)]$$

Theorem 1: The throughput for FAMA-LCS in a fully-connected network is given by

$$S = \frac{\delta B}{e^{\lambda\tau}(\frac{1}{\lambda} + \tau) + (A + \frac{1}{\lambda} + 2\gamma + \delta + 5\tau)B + (A + \frac{1}{\lambda})(e^{\lambda(\eta+\tau)} - B)} \quad (10)$$

Proof: Because the arrival of RTSs is Poisson with parameter λ , type-0 transmission periods are exponentially distributed and $T_0 = \frac{1}{\lambda}$. A type-2 transmission period consists of multiple RTSs starting at the beginning of the period, and can also contain additional RTSs that arrive to the channel within the vulnerability period of the period; therefore, $T_2 = \gamma + \tau + \bar{Y}$ seconds. The average value of Y is the same as in CSMA and equals [11] $\bar{Y} = \tau - \frac{1-e^{-\lambda\tau}}{\lambda}$. Therefore,

$$T_2 = \gamma + 2\tau - \frac{1 - e^{-\lambda\tau}}{\lambda} \quad (11)$$

A type-1 transmission period always contains an RTS and the associated propagation delay. If no RTSs arrive within τ seconds from the start of the transmission period, the period is successful and includes in addition a CTS, a data packet, an ACK, and the associated propagation delays. A type-1 transmission period is successful when no RTSs arrive within τ seconds from the start of the period, which also means that the first and the last RTS that arrives within τ seconds of its start are the same. Because RTS arrivals are Poisson with parameter λ and a CTS lasts $\gamma + 2\tau$, we obtain:

$$T_1 = \gamma + 2\tau - \frac{1 - e^{-\lambda\tau}}{\lambda} + e^{-\lambda\tau}(2\gamma + \delta + 5\tau) \quad (12)$$

The average utilization period in FAMA-LCS always lasts δ seconds. Therefore, because a type-1 transmission period succeeds with probability $e^{\lambda\tau}$, we have $U = e^{-\lambda\tau}\delta$ and the theorem follows by substituting the average values of transmission and utilization periods in Eq. 9. *Q.E.D.*

Theorem 2: The throughput for RIMA-SPL with $\xi = \tau$ in a fully-connected network is given by

$$S = \left(\delta B \frac{1}{N} \right) / \left[(A + \frac{1}{\lambda} + \frac{1}{N}(\gamma + \delta + 2\tau))B + e^{\lambda\tau}(\frac{1}{\lambda} + \tau) + (A + \frac{1}{\lambda})(e^{\lambda(\eta+\tau)} - B) \right] \quad (13)$$

Proof: Because the arrival of RTRs is Poisson with parameter λ , type-0 transmission periods are exponentially distributed and $T_0 = \frac{1}{\lambda}$. The average length of type-2 transmission periods is the same as in FAMA-LCS and given in Eq. (11), given that RTRs and RTSs last γ seconds.

A type-1 transmission period in RIMA-SPL always contains an RTR, the associated propagation delay, and the collision-avoidance waiting time, all of which lasts $\gamma + \tau + \xi$. When no RTRs arrive within τ seconds from the start of the transmission period, the period is successful if the polled node has a packet ready for the polling node; this happens with probability $e^{\lambda\tau}/N$. In this case the period also includes a data packet, an ACK, and two propagation delays. Therefore,

$$T_1 = \gamma + \xi + 2\tau - \frac{1 - e^{-\lambda\tau}}{\lambda} + \frac{e^{-\lambda\tau}}{N}(\delta + \gamma + 2\tau) \quad (14)$$

The average utilization period of RIMA-SPL always lasts δ seconds. An RTR succeeds in obtaining a data packet from a polled node if no other RTRs are sent within τ seconds of its start time and the polled node has data to send to the polling node; this probability equals $e^{-\lambda\tau}/N$. Therefore, $U = \delta e^{-\lambda\tau}/N$. The theorem follows by substituting the average values of transmission and utilization periods in Eq. 9. *Q.E.D.*

Theorem 3: The throughput of RIMA-DPL with $\xi > \gamma + 7\tau$ in a fully connected network is given by

$$S = \left((2e^{-\lambda\tau} - \frac{1}{N})\delta B \right) / \left[e^{\lambda\tau}(\frac{1}{\lambda} + \tau) + (A + \frac{1}{\lambda})(e^{\lambda(\eta+\tau)} - B) + (A + \frac{1}{\lambda} + e^{\lambda\tau}(2\delta + 2\gamma + 3\tau) - \frac{1}{N}(\delta + 2\gamma + 8\tau))B \right] \quad (15)$$

Proof: As in RIMA-SPL, the average length of type-0 transmission periods is $T_0 = \frac{1}{\lambda}$ and the average length of type-2 transmission periods is given by Eq. (11).

A type-1 transmission period in RIMA-SPL always contains an RTR, the associated propagation delay, and the collision-avoidance waiting time, all of which lasts $\gamma + \tau + \xi$. If no RTRs arrive within τ seconds from the start of the transmission period, there are two mutually exclusive cases to consider. If the polled node has no data packet to send to the polling node, the transmission period also includes a CTS, a data packet, and the associated propagation delays. Alternatively, if the polled node has data to send to the polling node, the period also includes a collision-avoidance interval, two data packets, two ACKs, and three propagation delays given that the polling node sends its ACK and a packet in sequence. Therefore, we obtain

$$T_1 = \gamma + \xi + 2\tau - \frac{1 - e^{-\lambda\tau}}{\lambda} + \frac{e^{-\lambda\tau}}{N}(\delta + \gamma + 2\tau) + (1 - \frac{e^{-\lambda\tau}}{N})(2\delta + 2\gamma + \xi + 3\tau) \quad (16)$$

Given that a successful type-1 transmission period contains one data packet when the polled node has not data to send and

two data packets when it does, the length of the average utilization period in RIMA-DPL equals

$$U = \frac{e^{-\lambda\tau}}{N}(\delta) + (1 - \frac{e^{-\lambda\tau}}{N})(2\delta) \quad (17)$$

Eq. (15) is obtained by substituting the average values of transmission and utilization periods in Eq. 9. *Q.E.D.*

C. Performance Comparison

To compare the limited-persistence collision-avoidance protocols introduced in this paper, we introduce the variables listed in Table I. We assume a fully-connected network topology with a propagation delay of $1\mu\text{s}$, the channel data rate of 1 Mbps, and preamble and processing overhead are ignored for convenience. Data packets are assumed to consist of 500 bytes, and the RTRs, ACKs, and RTSs used in all protocols consist of 20 bytes. For the case of RIMA-DPL, a CTS is also 20 bytes, and for FAMA-LCS a CTS lasts one round-trip longer than an RTS.

Figs. 6, 7 and 8 plot the throughput of FAMA-LCS, RIMA-SPL, and RIMA-DPL, against the average offered load of RTSs or RTRs when the network consists of 10 nodes. The figures also show the non-persistent variants of the collision-avoidance protocols. The three figures illustrate that proper amounts of limited persistence make all the collision avoidance schemes more efficient. In all cases, a persistence time of just a fraction (e.g., one half) of the transmission time of a control packet (RTR or RTS) gives the best results. As should be expected, all the protocols analyzed achieve higher throughput at light loads and sustained throughput up to moderate average loads. More marked improvements are obtained in receiver-initiated collision avoidance strategies than in sender-initiated strategies. The best results are obtained with RIMA-DPL, in which case the throughput at light average offered loads is markedly higher than in the non-persistence strategy.

The reason why limited persistence improves the efficiency of collision-avoidance protocols is that it tends to eliminate idle-time periods in the channel at light average loads, because stations are allowed to persist in their attempt to acquire control of the channel after detecting an ongoing transmission. Furthermore, throughput remains higher than with non-persistence at moderate offered loads, because only a fraction of those RTSs or RTRs that become ready for transmission while the channel is busy are allowed to contend for the channel when the channel becomes idle.

Given that all contention-based protocols, including collision-avoidance protocols, should operate in regions where offered traffic loads are light to moderate, our analysis shows that introducing limited persistence together with back off strategies that reduce the average offered load as congestion starts to mount is the right approach to making collision-avoidance protocols more efficient.

IV. CONCLUSIONS

We have presented the first treatment of limited persistence in collision-avoidance protocols. Limited persistence consists

$a = \frac{\tau}{\delta}$ (normalized propagation delay)
$b = \frac{\gamma}{\delta}$ (normalized control packets)
$h = \frac{\eta}{\delta}$ (normalized persistence duration)
$G = \lambda \times \delta$ (Offered Load, normalized to data packets)

TABLE I
NORMALIZED VARIABLES

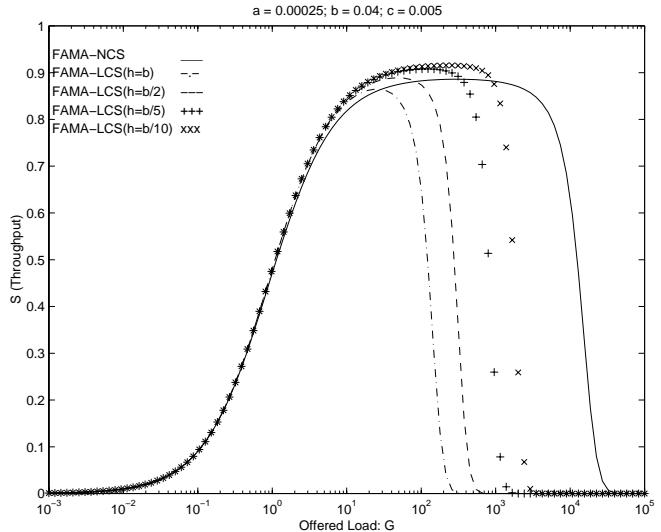


Fig. 6. Throughput vs. offered load for 1Mbit/sec channel and 500 Byte data packets for FAMA-LCS with different persistence intervals; network of 10 nodes

of allowing a station that detects a busy channel when it receives a packet to send to persist in engaging in a collision-avoidance dialogue for a limited amount of time only. The station is forced to back off for a random amount of time if the channel is busy or another station acquires control of the channel at the end of the persistence time.

Our analysis of limited persistence is based on earlier work on 1-persistent CSMA by Sohraby et al. [8]. Although this analysis assumes a fully-connected network, our results can be extrapolated to networks with hidden terminals, because RIMA and FAMA protocols provide correct collision avoidance, which means that the relative performance differences among these protocols observed in the analysis apply also to networks with hidden terminals.

Our analysis results show that a small persistence time of only a fraction of the transmission time of a collision-avoidance control packet (RTR or RTS) suffices to provide much higher throughput at high to moderate average offered loads. The performance improvement observed with limited persistence stems from reducing idle time in the channel due nodes backing off for large periods of time, and limiting the number of nodes that can contend for channel control after the channel becomes idle.

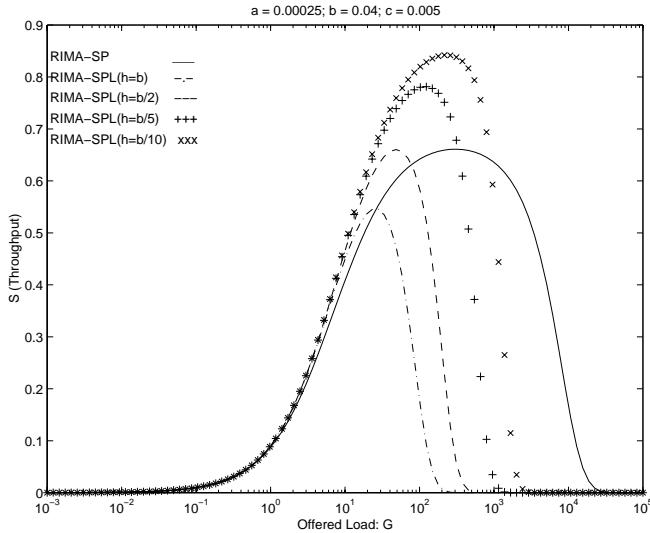


Fig. 7. Throughput vs. offered load for 1Mbit/sec channel and 500 Byte data packets for RIMA-SPL with different persistence intervals; network of 10 nodes

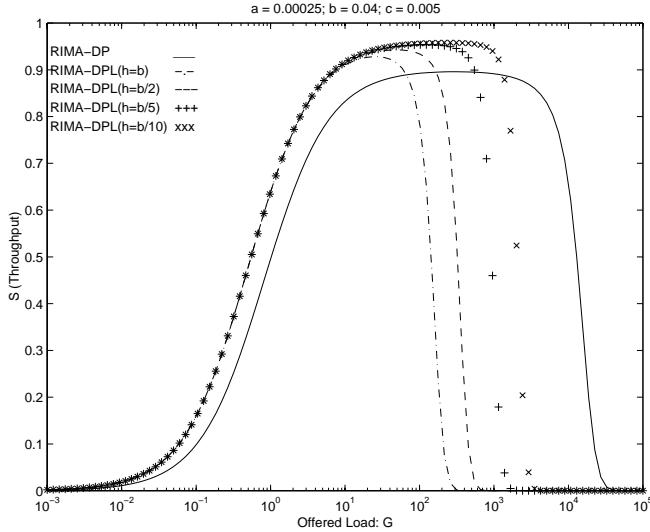


Fig. 8. Throughput vs. offered load for 1Mbit/sec channel and 500 Byte data packets for RIMA-DPL with different persistence intervals; network of 10 nodes

REFERENCES

- [1] IEEE P802.11 Draft Standard for Wireless LAN: Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE, July 1996.
- [2] C. L. Fullmer and J. J. Garcia-Luna-Aceves. Floor acquisition multiple access for packet-radio networks. In *Proceedings ACM SIGCOMM*, Cambridge, MA, September 1995.
- [3] C. L. Fullmer and J. J. Garcia-Luna-Aceves. Solutions to hidden terminal problems in wireless networks. In *Proceedings ACM SIGCOMM*, Cannes, France, September 1997.
- [4] J. J. Garcia-Luna-Aceves and A. Tzamaloukas. Receiver initiated collision-avoidance protocols for wireless networks. *Invited to appear in the ACM journal Wireless Networks (WINET), Special Issue on ad-hoc networks*, 1999.
- [5] J. J. Garcia-Luna-Aceves and A. Tzamaloukas. Reversing the collision-avoidance handshake in wireless networks. *Proceedings ACM/IEEE MOBICOM '99*, August 1999.
- [6] P. Karn. MACA - a new channel access method for packet radio. In *Proceedings ARRL/CRRL Amateur Radio 9th Computer Networking Conference*, New York, April 1990.
- [7] L. Kleinrock and F. A. Tobagi. Packet switching in radio channels: Part I - carrier sense multiple-access modes and their throughput-delay characteristics. *IEEE Transactions on Communications*, 23(12):1400–16, December 1975.
- [8] K. Sohraby, M. L. Molle, and A. N. Venetsanopoulos. Comments on throughput analysis for persistent csma systems. *IEEE Transactions on Communications*, COM-31:123–129, January 1987.
- [9] F. Talucci, M. Gerla, and L. Fratta. MACA-BI (MACA by invitation) - a receiver oriented access protocol for wireless multihop networks. In *Proceedings IEEE PIMRC*, 1997.
- [10] F. A. Tobagi. Multiaccess protocols in packet communications systems. *IEEE Transactions on Communications*, 28(4):468–488, April 1980.
- [11] F. A. Tobagi and L. Kleinrock. Packet switching in radio channels: Part II - the hidden terminal problem in carrier sense multiple-access modes and the busy-tone solution. *IEEE Transactions on Communications*, 23(12):1417–33, December 1975.
- [12] F. A. Tobagi and L. Kleinrock. Packet switching in radio channels: Part III - polling and (dynamic) split channel reservation multiple access. *IEEE Transactions on Communications*, 24(7):832–45, August 1976.